

Robust backstepping control for highly demanding quadrotor flight

Manuel Alejandro Vallejo-Alarcón, Martín Velasco-Villa,
Rafael Castro-Linares

*CINVESTAV-IPN, Department of Electrical Engineering, Mechatronics Section,
Av. IPN 2508, CP 07360, Mexico City, Mexico
(e-mail: {mvallejoa, velasco, rcastro }@cinvestav.mx)*

Abstract: This paper presents a control strategy that uses smoothly-bounded error corrections together with force-compensating terms to overcome perturbation problems that appear in quadrotor autonomous navigation, particularly in highly demanding flight conditions and slowly-varying wind conditions. First, the quadrotor dynamic model is reduced using feedback linearization. Then an integral-backstepping-like controller is designed, where bounded error-correction actions are introduced using smooth functions. Numerical simulations and real-time experiments are carried out to evaluate the proposed control strategy, showing an adequate behavior under high-acceleration trajectory tracking and slowly-varying wind conditions.

Keywords: Backstepping, UAVs, Robust control, Quadrotor.

1. INTRODUCTION

The quadrotor is an Unmanned Aerial Vehicle (UAV) that is able of making hover, Vertical Take-Off and Landing (VTOL), as well as omnidirectional flight. These characteristics make it appropriate for many applications. It also has a relatively well-known model (García-Carrillo et al., 2013); thus, it has become a widespread research platform. Using quadrotors for aggressive maneuvers or high-speed flights (Mellinger et al., 2012; Chen and Pérez-Arancibia, 2016; Cisneros et al., 2016; Liu et al., 2016; Lopez and How, 2017; Garcia et al., 2015; Pretorius and Boje, 2014; Dong et al., 2016) is useful in many ways, and with this interest becomes the need for control laws able to maintain stability under turbulence from highly-demanding trajectory planning. Also, it is needed to consider those perturbations that come from interactions with other UAVs or wind conditions that, by their nature, are non-vanishing perturbations.

Control laws designed using the backstepping technique (Sepulchre et al., 1997; Bouabdallah and Siegwart, 2005; Madani and Benallegue, 2006) show some robustness against exponentially bounded vanishing perturbations, so it is reasonable to design backstepping control laws that also present robustness against bounded non-vanishing perturbations, this can be achieved using techniques like the integral backstepping variation (Kanellakopoulos and Krein, 1993; Skjetne and Fossen, 2004; Bouabdallah and Siegwart, 2007; Vallejo-Alarcón et al., 2016).

Tracking highly-demanding trajectories can make grow the tracking errors significantly, so using linear error-correction actions can lead to control signals that can overpass the UAV physical capabilities, so it is necessary to use bounded error-correction actions. The control signals can be bounded in final implementations through software adjustments, but this can cause an inappropriate mathematical representation of the

control laws, thus it is necessary to propose control laws that include bounded error-correction actions. The literature has results about bounded control, in general terms as in (Loria and Nijmeijer, 1998) where it is presented for fully actuated Euler-Lagrange systems, or applied to quadrotors using non-smooth bounds as in (Guerrero-Castellanos et al., 2011; Cheng et al., 2017; Huang et al., 2017) or using smooth bounds as in (Guadarrama-Olvera et al., 2014).

In (Vallejo-Alarcón et al., 2016) an integral-backstepping-like control law design for a quadrotor is presented. The control law is robust when modeled bounds over the error-correction actions are considered allowing to have maximum translational and rotational speeds, and accelerations, due to error corrections; robustness is accomplished using extended states to provide integral actions, and the error-correction bounds are provided using the hyperbolic tangent function. Also, simulation results are presented. However, that control law design presents a Real-Time (RT) implementation inconvenience due ground-air communication requirements and embedded computer processing availability. In this present paper, the control law is redesigned forcing the backstepping steps to set, using desired angles, thus reducing UAV communication and computer processing requirements. The proposed control law is tested through numerical simulations and RT experimentation.

The paper is organized as follows. Section 2 presents the quadrotor dynamic model and a feedback linearization. In Section 3, it is presented a backstepping control law design for the feedback linearized quadrotor model, using smoothly-bounded error correction and integral terms. In Section 4, numerical simulations are carried out to show performance and viability, then in Section 5 RT experimentation results are presented. Finally, Section 6 presents the main conclusions of the work.

2. QUADROTOR DYNAMIC MODEL

The quadrotor dynamic model (García-Carrillo et al., 2013) on the North-East-Down (NED) inertial frame, with the position $\xi = [x \ y \ z]^T$, and the so-called Euler angles $\eta = [\phi \ \theta \ \psi]^T$ (ϕ is the roll, θ is the pitch, and ψ is the yaw), perturbed by a force vector $\delta_\xi = [\delta_{x_1} \ \delta_{y_1} \ \delta_{z_1}]^T$ and a torque vector $\delta_\eta \in R^3$, is given by

$$\begin{aligned} m\ddot{\xi} &= mge_3 + Rf - \delta_\xi, \\ J\dot{\Omega} &= J\Omega \times \Omega + \tau - \delta_\eta, \end{aligned} \quad (1)$$

where \times denotes the cross product of vectors and $\Omega = [p \ q \ r]^T$ is the angular speed in the body fixed frame (XYZ), as shown in Fig. 1. Additionally, m stands for the quadrotor mass, g is the gravitational acceleration constant, and e_3 is the unitary vector in the direction of the D axis. Also, the matrix R , represented as

$$R = \begin{bmatrix} C_\theta C_\psi & S_\phi S_\theta C_\psi - C_\phi S_\psi & C_\phi S_\theta C_\psi + S_\phi S_\psi \\ C_\theta S_\psi & S_\phi S_\theta S_\psi + C_\phi C_\psi & C_\phi S_\theta S_\psi - S_\phi C_\psi \\ -S_\theta & S_\phi C_\theta & C_\phi C_\theta \end{bmatrix}$$

associates the inertial and the body-fixed frames, where the compact notation $S_\gamma = \mathbf{sin}(\gamma)$, $C_\gamma = \mathbf{cos}(\gamma)$, and $T_\gamma = \mathbf{tan}(\gamma)$ is used (this notation is going to be used henceforth). Furthermore, $J = \mathbf{diag}\{J_{x_x}, J_{y_y}, J_{z_z}\}$ is the inertia matrix, $\tau = [\tau_\phi \ \tau_\theta \ \tau_\psi]^T$ is the generalized torque, and $f = [0 \ 0 \ -u]^T$ is the total force applied to the quadrotor, with $u = r_{t_1} + r_{t_2} + r_{t_3} + r_{t_4}$ the total thrust and r_{t_i} being the i th rotor thrust.

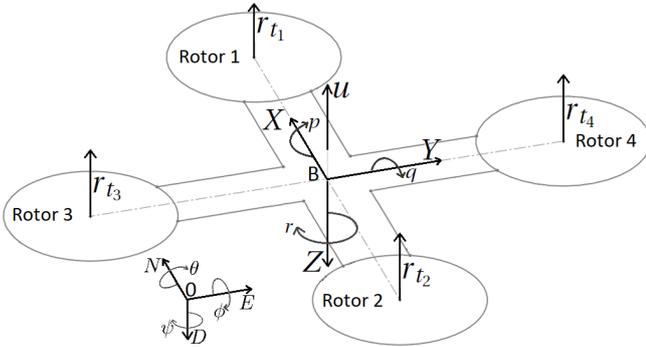


Fig. 1. Quadrotor in the NED inertial frame.

Using the expression $\Omega = W\dot{\eta}$, that relates the angular speed in the body-fixed frame with the time derivative of the Euler angles, where

$$W = \begin{bmatrix} 1 & 0 & -S_\theta \\ 0 & C_\phi & C_\theta S_\phi \\ 0 & -S_\phi & C_\theta C_\phi \end{bmatrix},$$

one has that $\dot{\Omega} = \dot{W}\dot{\eta} + W\ddot{\eta}$. This allows to rewrite the quadrotor dynamic model (1) as

$$\begin{aligned} \begin{bmatrix} \ddot{x} \\ \ddot{y} \end{bmatrix} &= -\frac{u}{m} \begin{bmatrix} C_\psi & S_\psi \\ S_\psi & -C_\psi \end{bmatrix} \begin{bmatrix} S_\theta C_\phi \\ S_\phi \end{bmatrix} - \frac{1}{m} \begin{bmatrix} \delta_{x_1} \\ \delta_{y_1} \end{bmatrix}, \\ \ddot{z} &= g - \frac{u}{m} C_\theta C_\phi - \frac{\delta_{z_1}}{m}, \\ W^T J W \dot{\eta} &= W^T (J W \dot{\eta} \times W \dot{\eta}) - W^T J \dot{W} \dot{\eta} + W^T \tau - W^T \delta_\eta. \end{aligned} \quad (2)$$

It is important to notice that the dynamics (2) present inherent limitations in relation with the Euler angles, from the point of view of singularities when the input variable is the total thrust u . So, it is considered that the following assumption holds.

Assumption 1. The Euler angles θ and ϕ are restricted to evolve in accordance to $|\theta| < \pi/2$ and $|\phi| < \pi/2$.

Such an assumption, on the dynamics of the angles θ and ϕ , is not a restrictive one since most real flying maneuvers for quadrotors are limited to these values in the Euler angles. Moreover, as an implementation-only safety measure, the desired values for θ and ϕ were programmatically restricted to not overpass $\pm\pi/3$, for numerical simulation as well as for RT experimentation; however, this safety feature was not performed since those values were never reached in the present results.

Based on the nonlinear dynamic inversion approach, the following preliminary inputs are chosen,

$$u = \frac{m(g - \tilde{u})}{C_\theta C_\phi}, \quad (3)$$

$$\tau = -JW\dot{\eta} \times W\dot{\eta} + J\dot{W}\dot{\eta} + JW\tilde{\tau},$$

where \tilde{u} and $\tilde{\tau} = [\tilde{\tau}_\phi \ \tilde{\tau}_\theta \ \tilde{\tau}_\psi]^T$ are new inputs. Inaccurate values for J and m would lead to adding input-dependent perturbations, so they are assumed to be accurate enough; otherwise, their perturbation effects could be included in the perturbations already expressed. Then, one has the partially linear model

$$\begin{aligned} \begin{bmatrix} \ddot{x} \\ \ddot{y} \end{bmatrix} &= (\tilde{u} - g) \begin{bmatrix} C_\psi & S_\psi \\ S_\psi & -C_\psi \end{bmatrix} \begin{bmatrix} T_\theta \\ T_\phi / C_\theta \end{bmatrix} - \begin{bmatrix} \delta_x \\ \delta_y \end{bmatrix}, \\ \ddot{z} &= \tilde{u} - \delta_z, \\ \ddot{\phi} &= \tilde{\tau}_\phi - \delta_\phi, \\ \ddot{\theta} &= \tilde{\tau}_\theta - \delta_\theta, \\ \ddot{\psi} &= \tilde{\tau}_\psi - \delta_\psi, \end{aligned} \quad (4)$$

with $[\delta_\phi \ \delta_\theta \ \delta_\psi]^T = W^{-1}J^{-1}\delta_\eta$ and $[\delta_x \ \delta_y \ \delta_z]^T = m^{-1}\delta_\xi$.

3. CONTROL LAW DESIGN

The partially linearized model (4) is considered to design a control law. To synthesize the feedback law, the states of (4) are defined as

$$\begin{aligned} x_1 &= [z \ \psi]^T, & x_2 &= [\dot{z} \ \dot{\psi}]^T, \\ x_3 &= [x \ y]^T, & x_4 &= [\dot{x} \ \dot{y}]^T, \\ x_5 &= [\theta \ \phi]^T, & x_6 &= [\dot{\theta} \ \dot{\phi}]^T, \end{aligned}$$

the inputs are redefined in vectors as

$$u_1 = [\tilde{u} \ \tilde{\tau}_\psi]^T, u_2 = [\tilde{\tau}_\theta \ \tilde{\tau}_\phi]^T,$$

and the perturbation vectors as

$$\delta_1 = [\delta_z \ \delta_\psi]^T, \delta_2 = [\delta_x \ \delta_y]^T, \delta_3 = [\delta_\theta \ \delta_\phi]^T.$$

Under these conditions, model (4) can be written as

$$\begin{aligned} \dot{x}_1 &= x_2, \\ \dot{x}_2 &= u_1 - \delta_1, \\ \dot{x}_3 &= x_4, \\ \dot{x}_4 &= G\varphi(x_5) - \delta_2, \\ \dot{x}_5 &= x_6, \\ \dot{x}_6 &= u_2 - \delta_3, \end{aligned} \quad (5)$$

with

$$G = (\tilde{u} - g) \begin{bmatrix} C_\psi & S_\psi \\ S_\psi & -C_\psi \end{bmatrix}, \varphi(x_5) = \begin{bmatrix} T_\theta \\ T_\phi / C_\theta \end{bmatrix}.$$

3.1 Design by integral-backstepping-like approach

Defining a set of desired state-vectors as

$$\begin{aligned} x_{1_d} &= [z_d(t) \ \psi_d(t)]^T, \\ x_{3_d} &= [x_d(t) \ y_d(t)]^T, \\ x_{5_d} &= [\theta_d \ \phi_d]^T, \end{aligned}$$

where $x_d(t)$, $y_d(t)$, $z_d(t)$, $\psi_d(t)$, θ_d , and ϕ_d are the desired values of x , y , z , ψ , θ , and ϕ respectively. It is important to point out that $h(t) = [x_d(t) \ y_d(t) \ z_d(t) \ \psi_d(t)]^T$ is a class C^4 vector function that defines the desired trajectory; θ_d and ϕ_d are to be obtained later. Also, defining the tracking errors

$$e_1 = [e_z \ e_\psi]^T = x_{1_d} - x_1,$$

$$e_3 = [e_x \ e_y]^T = x_{3_d} - x_3,$$

$$e_5 = [e_\theta \ e_\phi]^T = x_{5_d} - x_5,$$

and considering the dynamics (5), one gets

$$\dot{e}_1 = \dot{x}_{1_d} - x_2, \quad (6a)$$

$$\dot{e}_3 = \dot{x}_{3_d} - x_4, \quad (6b)$$

$$\dot{e}_5 = \dot{x}_{5_d} - x_6. \quad (6c)$$

Following the backstepping approach, starting from (6a), x_2 is considered to be the virtual input $v_1 = [v_z \ v_\psi]^T$ of the subsystem (6a), more specifically

$$\dot{e}_1 = \dot{x}_{1_d} - v_1, \quad (7a)$$

and the tracking error e_2 is defined as

$$e_2 = [e_z \ e_\psi]^T = v_1 - x_2.$$

Thus, the following subsystem is obtained through derivation:

$$\dot{e}_2 = \dot{v}_1 - u_1 + \delta_1. \quad (7b)$$

Using similar arguments for (6b), x_4 is considered to be the virtual input $v_3 = [v_x \ v_y]^T$, thus,

$$\dot{e}_3 = \dot{x}_{3_d} - v_3, \quad (7c)$$

and the tracking error e_4 is defined as

$$e_4 = [e_x \ e_y]^T = v_3 - x_4.$$

Then, through derivation, it is obtained the subsystem

$$\dot{e}_4 = \dot{v}_3 - G\varphi(x_5) + \delta_2.$$

Additionally, $\varphi(x_5)$ is considered to be the virtual input $v_4 = [v_x \ v_y]^T$, thus,

$$\dot{e}_4 = \dot{v}_3 - Gv_4 + \delta_2. \quad (7d)$$

In order, to obtain the value of x_{5_d} , it is necessary to notice that v_4 serves as the desired value of $\varphi(x_5)$, namely $\varphi(x_{5_d})$, so,

$$v_4 = [v_x \ v_y]^T = \varphi(x_{5_d}) = [\tan \theta_d \ \tan \phi_d / \cos \theta_d]^T.$$

Consequently

$$\theta_d = \arctan(v_x),$$

$$\phi_d = \arctan\left(v_y \cos(\arctan(v_x))\right).$$

Since, it is known that

$$\cos(\arctan v_x) = 1 / \sqrt{1 + v_x^2},$$

one has that

$$x_{5_d}(v_4) = \begin{bmatrix} \theta_d(v_4) \\ \phi_d(v_4) \end{bmatrix} = \begin{bmatrix} \arctan(v_x) \\ \arctan\left(\frac{v_y}{\sqrt{1 + v_x^2}}\right) \end{bmatrix}.$$

Subsequently, from (6c), it is considered that x_6 is the virtual input $v_5 = [v_\theta \ v_\phi]^T$, so,

$$\dot{e}_5 = \dot{x}_{5_d} - v_5, \quad (7e)$$

and tracking error e_6 is defined as

$$e_6 = \begin{bmatrix} e_\theta \\ e_\phi \end{bmatrix} = v_5 - x_6.$$

Then, it is obtained the last subsystem through derivation as

$$\dot{e}_6 = \dot{v}_5 - u_2 + \delta_3. \quad (7f)$$

So, the system model, under error states, is represented by equations (7). Then considering the perturbations of system (2) as non-vanishing and bounded, and noticing that those perturbations affect the model (7) through equations (7b), (7d), and (7f), the states, $\hat{\beta}_1 = K_1 e_2$, $\hat{\beta}_2 = K_2 e_4$, and $\hat{\beta}_3 = K_3 e_6$ are defined with being $K_{1,2,3} > \mathbf{0} \in \mathbb{R}^{2 \times 2}$ diagonal matrices; such a state definition allows to measure the effects of the perturbations on the tracking error dynamics. The extended model is then given by

$$\begin{aligned}
\dot{e}_1 &= \dot{x}_{1d} - v_1, \\
\dot{e}_2 &= \dot{v}_1 - u_1 + \delta_1, \\
\dot{e}_3 &= \dot{x}_{3d} - v_3, \\
\dot{e}_4 &= \dot{v}_3 - Gv_4 + \delta_2, \\
\dot{e}_5 &= \dot{x}_{5d}(v_4) - v_5, \\
\dot{e}_6 &= \dot{v}_5 - u_2 + \delta_3, \\
\dot{\beta}_1 &= K_1 e_2, \quad \dot{\beta}_2 = K_2 e_4, \quad \dot{\beta}_3 = K_3 e_6,
\end{aligned} \tag{8}$$

where the following assumption is made on the perturbation terms δ_i and its time derivatives $\dot{\delta}_i$, and the terms $\alpha_i = \delta_i - \beta_i$, for $i = 1, 2, 3$.

Assumption 2. The perturbations and errors are bounded as $|\delta_i| \leq \gamma_i$, $|\dot{\delta}_i| \leq \lambda_i$, and $|\alpha_i| \leq \zeta_i$, with $\gamma_i, \lambda_i, \zeta_i > 0 \in \mathbb{R}^2$.

In order to meet such an assumption, notice that $|\alpha_i| \leq \gamma_i + |\beta_i|$, then, as β_i depends on $e_{(2i)}$, and these in system inputs and bounded disturbances, it is necessary to have bounded system inputs. This condition is true since the UAV capabilities are limited.

One now states the following result that allows to keep the tracking errors evolving in a bounded region around zero under a sufficient condition on the perturbation and corresponding bounded error.

Theorem 1. The quadrotor model (2), under the linearization (3) and the error-states extended representation (8) in the region where Assumption 1 holds, is locally asymptotically stabilized to track a class \mathbf{C}^4 trajectory defined by $\mathbf{h}(\mathbf{t}) = [\mathbf{x}_d(\mathbf{t}) \mathbf{y}_d(\mathbf{t}) \mathbf{z}_d(\mathbf{t}) \Psi_d(\mathbf{t})]^T$ if the perturbations meet $\dot{\delta}_i = \mathbf{0}$, $\mathbf{i} \in \{1, 2, 3\}$, and is kept under a practical stability margin if the Assumption 2 holds, by means the control law

$$\begin{aligned}
v_1 &= \dot{x}_{1d} + A_{11} \tanh(A_{12} e_1), \\
u_1 &= \dot{v}_1 + A_{21} \tanh(A_{22} e_2) + \beta_1, \\
v_3 &= \dot{x}_{3d} + A_{31} \tanh(A_{32} e_3), \\
v_4 &= G^{-1}(\dot{v}_3 + A_{41} \tanh(A_{42} e_4) + \beta_2), \\
v_5 &= \dot{x}_{5d} + A_{51} \tanh(A_{52} e_5), \\
u_2 &= \dot{v}_5 + A_{61} \tanh(A_{62} e_6) + \beta_3,
\end{aligned} \tag{9}$$

with $A_{ij} > 0 \in \mathbb{R}^{2 \times 2}$, diagonal matrices, for $i = 1, \dots, 6$ and $j = 1, 2$.

Proof. Considering (8) and applying (9), it is obtained

$$\begin{aligned}
\dot{e}_1 &= -A_{11} \tanh(A_{12} e_1), \\
\dot{e}_2 &= -A_{21} \tanh(A_{22} e_2) + \delta_1 - \beta_1, \\
\dot{e}_3 &= -A_{31} \tanh(A_{32} e_3), \\
\dot{e}_4 &= -A_{41} \tanh(A_{42} e_4) + \delta_2 - \beta_2, \\
\dot{e}_5 &= -A_{51} \tanh(A_{52} e_5), \\
\dot{e}_6 &= -A_{61} \tanh(A_{62} e_6) + \delta_3 - \beta_3, \\
\dot{\beta}_1 &= K_1 e_2,
\end{aligned}$$

$$\dot{\beta}_2 = K_2 e_4,$$

$$\dot{\beta}_3 = K_3 e_6,$$

where are two kinds of isolated scalar subsystems,

$$\dot{e}_a = -a_1 \tanh(a_2 e_a), \tag{10}$$

and

$$\begin{aligned}
\dot{e}_b &= -b_1 \tanh(b_2 e_b) + \alpha_b, \\
\dot{\beta}_b &= k_b e_b.
\end{aligned} \tag{11}$$

For (10) it is proposed the Lyapunov function candidate

$$V_a(e_a) = \frac{1}{2} e_a^2,$$

so,

$$\dot{V}_a = e_a \dot{e}_a,$$

$$\dot{V}_a = -a_1 e_a \tanh(a_2 e_a) < 0.$$

For (11) it is proposed the Lyapunov function candidate

$$V_b(e_b, \alpha_b) = \frac{1}{2} e_b^2 + \frac{1}{2k_b} \alpha_b^2,$$

then

$$\begin{aligned}
\dot{V}_b &= e_b \dot{e}_b + \frac{1}{k_b} \alpha_b \dot{\alpha}_b \\
&= e_b (-b_1 \tanh(b_2 e_b) + \alpha_b) + \frac{1}{k_b} \alpha_b (\dot{\delta}_b - \dot{\beta}_b) \\
&= -b_1 e_b \tanh(b_2 e_b) + e_b \alpha_b + \frac{1}{k_b} \alpha_b (\dot{\delta}_b - k_b e_b) \\
&= -b_1 e_b \tanh(b_2 e_b) + \frac{\alpha_b \dot{\delta}_b}{k_b},
\end{aligned} \tag{12}$$

so, if $\dot{\delta}_b = \mathbf{0}$

$$\dot{V}_b = -b_1 e_b \tanh(b_2 e_b) \leq 0,$$

then considering that the only set where $\dot{V}_b = 0$ is $e_b \equiv 0$, that implies $\dot{e}_b \equiv 0$, and, from (11), one has that $\alpha_b = 0$, so the LaSalle's invariance principle is met. Thus, the subsystem is said to be locally asymptotically stable when the perturbation δ_b is constant.

Then, from (12),

$$\begin{aligned}
\dot{V}_b &= -b_1 e_b \tanh(b_2 e_b) + \frac{\alpha_b \dot{\delta}_b}{k_b} \\
&\leq -b_1 e_b \tanh(b_2 e_b) + \frac{|\alpha_b| |\dot{\delta}_b|}{k_b} \\
&\leq -b_1 e_b \tanh(b_2 e_b) + \frac{\zeta_b \lambda_b}{k_b},
\end{aligned}$$

so, when the sufficient condition,

$$\zeta_b \lambda_b \leq b_1 k_b e_b \tanh(b_2 e_b)$$

is met the system is said to be under practical stability. ■

Remark 1. To fulfill the sufficient condition, it is necessary that either the bounds $\zeta_{\mathbf{b}}$ or $\lambda_{\mathbf{b}}$ be sufficiently small or the error $\mathbf{e}_{\mathbf{b}}$ large enough, these magnitudes are affected by the control gains \mathbf{b}_1 , \mathbf{b}_2 , and $\mathbf{k}_{\mathbf{b}}$, so making the control gains large turns the necessary error, to keep the system stable, arbitrarily small.

Remark 2. Due to a singularity, in \mathbf{G}^{-1} when $\tilde{\mathbf{u}} = \mathbf{g}$, it is assumed that the control law (9) cannot impose free-fall acceleration in \mathbf{z} .

Remark 3. The control law (9) imposes maximum values to the error-correction actions by means of the \mathbf{A}_{i1} matrices, also giving a desired growth rate to the error-correction actions through the \mathbf{A}_{i2} matrices.

Remark 4. The control law (9) presents no direct limitations to the desired trajectory, but it produces necessity of trajectories that not overpass the actuators capabilities. Then, in order to perform a safe RT experimentation avoiding physical system saturations, the desired trajectories were chosen in such a manner to keep $|\theta| < \pi/4$ and $|\phi| < \pi/4$.

4. NUMERIC SIMULATION

Before conducting RT experimentation, simulation software was developed, and numerical simulations were carried out to evaluate the designed control law. This was made by taking into consideration physical restrictions and the capabilities of the experimental platform. Also, all software developments were made using the same programming language as the experimental platform uses, so the control law implementations, for simulation and RT experimentation, are equal.

4.1 Approximated model

Taking into account the hardware limitations for the RT experimentation, and the objective to keep the numerical simulation as close as possible to the RT experimentation, it is needed to reduce the computational effort to evaluate the feedback linearization (3). So, to design a feedback linearization that use less computational effort, one may consider that $\dot{\eta} \approx 0$ together with the zero order Taylor expansion of entries of the matrix W , for $\theta = 0$ and $\phi = 0$ (this is, W becomes the identity matrix $I_3 \in \mathbb{R}^{3 \times 3}$). These considerations allow to rewrite model (2) as

$$\begin{aligned} \begin{bmatrix} \ddot{x} \\ \ddot{y} \end{bmatrix} &= -\frac{u}{m} \begin{bmatrix} C_\psi & S_\psi \\ S_\psi & -C_\psi \end{bmatrix} \begin{bmatrix} S_\theta C_\phi \\ S_\phi \end{bmatrix} - \frac{1}{m} \begin{bmatrix} \delta_{x_1} \\ \delta_{y_1} \end{bmatrix}, \\ \ddot{z} &= g - \frac{u}{m} C_\theta C_\phi - \frac{\delta_{z_1}}{m}, \\ \ddot{\eta} &= J^{-1}\tau - \delta_\eta. \end{aligned} \quad (13)$$

Notice that this approximated model losses accuracy as the angle speeds $\dot{\eta}$ are away from zero (correspondingly, η is also varying and is away from zero). This model takes again the form (4) by means of the simplified feedback

$$\begin{aligned} u &= \frac{m(g - \tilde{u})}{C_\theta C_\phi}, \\ \tau &= J\tilde{\tau}, \end{aligned} \quad (14)$$

where, as above, $\tilde{\mathbf{u}}$ and $\tilde{\tau} = [\tilde{\tau}_\phi \ \tilde{\tau}_\theta \ \tilde{\tau}_\psi]^T$ are new input signals.

4.2 Simulation implementation

The quadrotor model used to simulate the system physics is (2). Then, reading the states from the physics simulation, the control law (9) is evaluated, to finally transform the control law output signals to system input signals using the simplified feedback linearization (14). Taking this into account, the simulation software was designed to run using three threads. The first one represents the ground station that reads the UAV states and partially evaluates the control law at 100Hz. The second thread represents the UAV embedded computer that finishes the control law evaluation at 1kHz. And the third one represents the physical system through numerical integration of the model at 10kHz. The Fig. 2 shows a scheme of the simulation program flow.

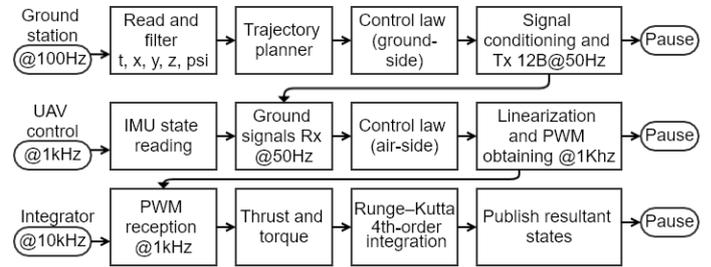


Fig. 2. Simulator scheme.

The considered quadrotor parameters are

$$m = 0.616\text{kg},$$

$$g = 9.81\text{m/s}^2,$$

$$J = \text{diag}\{3.605 \quad 3.721 \quad 7.031\} \times 10^{-3}\text{kg} \cdot \text{m}^2.$$

The elected bounds for the error-correction actions, imposed by the control law, are

$$A_{11} = \text{diag}[1\text{m/s}, 1.57\text{rad/s}] \quad \text{for vertical speed and yaw angular speed,}$$

$$A_{21} = \text{diag}[1.96\text{m/s}^2, 6.28\text{rad/s}^2] \quad \text{for vertical and yaw angular accelerations,}$$

$$A_{31} = \text{diag}[1\text{m/s}, 1\text{m/s}] \quad \text{for lateral speeds,}$$

$$A_{41} = \text{diag}[2\text{m/s}^2, 2\text{m/s}^2] \quad \text{for lateral accelerations,}$$

$$A_{51} = \text{diag}[50\text{rad/s}, 50\text{rad/s}] \quad \text{for pitch and roll angular speeds,}$$

$$A_{61} = \text{diag}[50\text{rad/s}^2, 50\text{rad/s}^2] \quad \text{for pitch and roll angular accelerations.}$$

The controller gains for the error-correction actions, designed considering their relations with the elected bounds of the control law, are given by

$$A_{12} = \text{diag}[2.5, 2.55],$$

$$A_{22} = \text{diag}[2.55, 0.64],$$

$$A_{32} = \text{diag}[3.2, 3.2],$$

$$A_{42} = \text{diag}[1.9, 1.9],$$

$$A_{52} = \text{diag}[0.6,0.6],$$

$$A_{62} = \text{diag}[0.6,0.6],$$

and the gains applied to the extended states, providing integral control actions, are given by

$$K_1 = \text{diag}[15,7], \quad K_2 = \text{diag}[1,1], \quad K_3 = \text{diag}[50,50].$$

4.3 Numeric simulation for a quadrotor tracking a high-acceleration trajectory

Numeric simulation results are presented considering the above settings. Also, to test the performance of the proposed control law, under the simulation environment, a desired trajectory was designed. This desired trajectory, represented in meters for x_d , y_d , and z_d , and in radians for ψ_d , is given by

$$\begin{aligned} x_d &= x_{bt} S_f(t, 20, 88, 0.3, 0.3), \\ y_d &= y_{bt} S_f(t, 21.25, 87.5, 0.6, 0.6), \\ z_d &= -\frac{1}{2} S_f(t, 5, 95, 0.95, 0.95), \\ \psi_d &= 0, \end{aligned} \quad (15)$$

with

$$x_{bt} = -\frac{1}{2}(1 + \sin(5t/\pi)),$$

$$y_{bt} = \frac{1}{2}\sin(10t/\pi),$$

and where the function

$$S_f(t, t_1, t_2, p_1, p_2) = \frac{(\tanh((t-t_1)p_1)+1)(1-\tanh((t-t_2)p_2))}{4} \quad (16)$$

is used to smoothly couple trajectory segments by activating or deactivating behaviors like x_{bt} for x_d or y_{bt} for y_d , with t as the time, t_1 and t_2 respectively as the starting and ending time of the coupled trajectory, and p_1 and p_2 as the coupling and decoupling rates; this way preventing the appearance of trajectory derivatives high enough to overpass the UAV capabilities.

4.3.1 Numeric simulation results

The Figures 3 to 7 were obtained using the proposed control law (9) and the desired trajectory (15), in the numerically simulated environment. In Fig. 3 are shown the desired and the simulated trajectories in the NED inertial frame, Fig. 4 shows the states x , y , z , and ψ compared to its desired counterparts, then, in Fig. 5 are shown the tracking errors for those states. Subsequently, Fig. 6 shows the desired and simulated values of θ and ϕ , and the errors between them. Later, in Fig. 7 are shown the obtained Pulse-Width Modulation (PWM) duty signals, where PWM_i was applied to the i th rotor in the simulated environment, additionally are shown the mean, maximum, and minimum PWM duties. Also, all presented error signals show their Mean Absolute Error (MAE) and Absolute Maximum Error (AME).

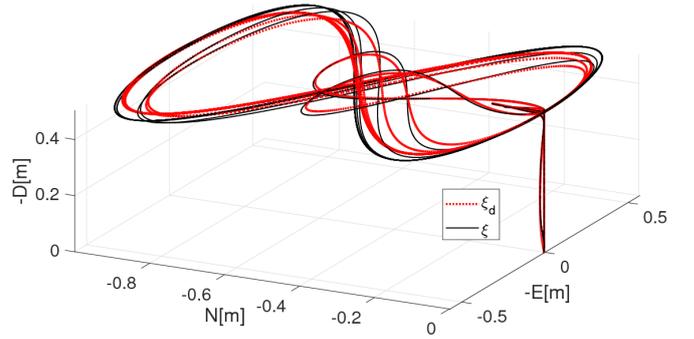


Fig. 3. Desired and simulated trajectories in NED frame.

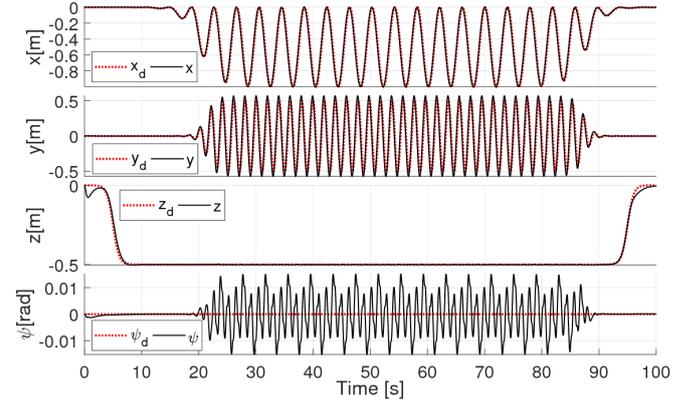


Fig. 4. Simulated and desired values of x , y , z , and ψ .

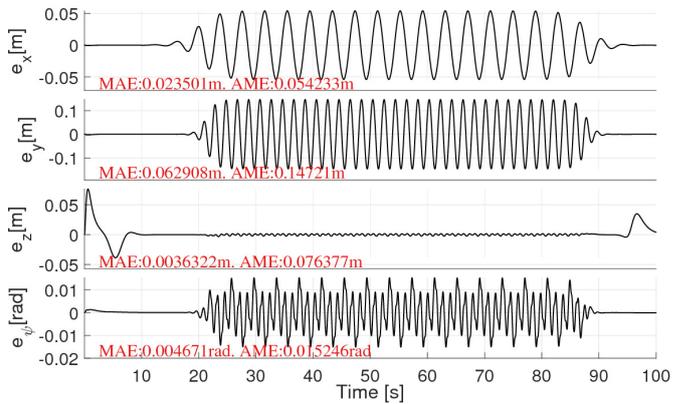


Fig. 5. Simulation tracking errors of x , y , z , and ψ .

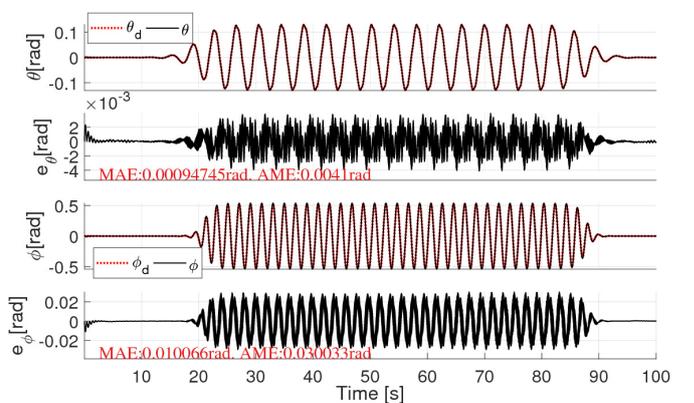


Fig. 6. Simulated and desired values, and tracking errors, of θ and ϕ .

4.3.2 Simulation discussion

In Figures 3 and 4 can be observed the simulated and desired trajectories for ξ , additionally for ψ in Fig. 4; in these figures the simulated trajectory closely tracks the desired one. So, as shown in Fig. 5, the tracking errors are kept bounded under cyclical acceleration, also must be pointed out that the height error e_z only grows in the takeoff and landing routines and is kept close to zero elsewhere.

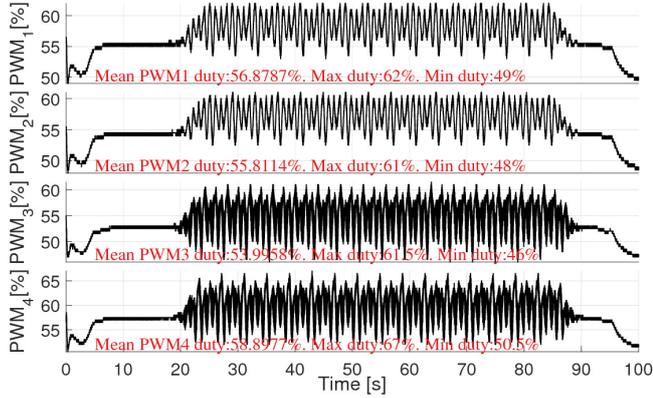


Fig. 7. PWM duties, applied to the UAV numerical model.

The angular errors e_ϕ and e_θ are maintained close to zero, as can be observed in Fig. 6, even with ϕ over $\pi/6$ rad, where the approximated model (13), used to calculate the control law, loses 13.39% of cosine accuracy and 50% of sine accuracy. Also, the applied PWM signals were kept under acceptable conditions for the physical system, staying close to 50% of PWM duty and never passing from 67%, as shown in Fig. 7.

4.4 Simulation for a perturbed quadrotor tracking a trajectory

Numeric simulation results are presented considering the settings in Subsection 4.2. To test the performance of the proposed control law, under the simulation environment, the system is disturbed by non-vanishing perturbations shown in Fig. 8 and defined as

$$\begin{aligned}\delta_z &= (0.5 + 0.05\sin(t/2\pi))S_f(t, 12,92,1.77,1.77), \\ \delta_\psi &= (3 + 0.3\sin(t/3\pi))S_f(t, 12,92,1.77,1.77), \\ \delta_x &= (0.25 + 0.025\sin(t/2.5\pi))S_f(t, 12,92,1.77,1.77), \\ \delta_y &= (0.2 + 0.02\sin(t/1.5\pi))S_f(t, 12,92,1.77,1.77), \\ \delta_\theta &= 0, \quad \delta_\phi = 0.\end{aligned}\quad (17)$$

Also, to test the system under the presented perturbations a desired trajectory was designed, this trajectory, represented in meters for x_d , y_d , and z_d , and in radians for ψ_d , is given by

$$\begin{aligned}x_d &= x_{pt}S_f(t, 20,88,0.3,0.3), \\ y_d &= y_{pt}S_f(t, 21.25,87.5,0.6,0.6), \\ z_d &= -S_f(t, 5,95,0.95,0.95)/2, \\ \psi_d &= 0,\end{aligned}\quad (18)$$

with

$$x_{pt} = -(1 + \sin(4t/\pi))/2, \quad y_{pt} = \sin(8t/\pi)/2,$$

where the function S_f is defined in (16).

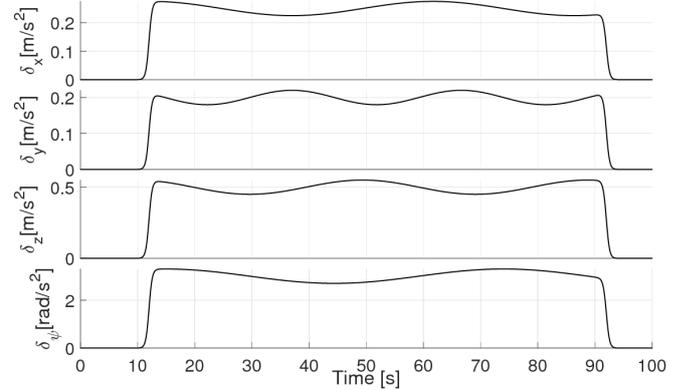


Fig. 8. Non-vanishing perturbations.

4.4.1 Simulation results

The Figures 9 to 15 were obtained using the proposed control law (9) and the desired trajectory (18) under the perturbations (17), in the numerically simulated environment. In Fig. 9 are shown the desired and simulated trajectories for the system under perturbations in the NED inertial frame. The Fig. 10 shows the desired and simulated trajectory components over time, then Fig. 11 shows the tracking errors between desired and simulated trajectory components. The Fig. 12 shows the in-simulation calculated desired values of θ and ϕ , their simulation values and the tracking errors between them. Fig. 13 shows the PWM duty signals applied to the simulated system. Finally, the Fig. 14 shows the applied non-vanishing perturbations δ and the extended states β , then in Fig. 15 are shown the errors between perturbations and extended states. Additionally, all presented error signals show their MAE and AME.

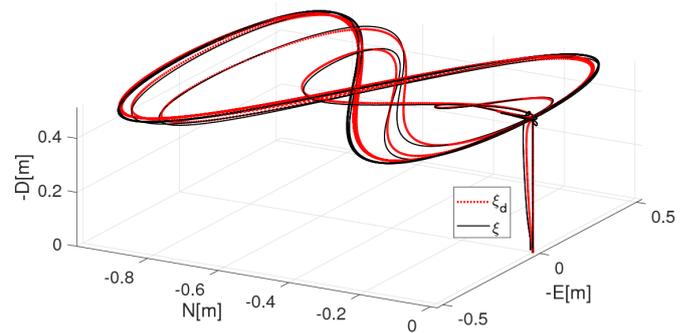


Fig. 9. Desired and simulated trajectories in **NED** frame.

4.4.2 Simulation discussion

The Figures 9 and 10 show the desired and simulated trajectories of the simulated perturbed quadrotor, where can be observed the simulated trajectory closely tracking the desired one, showing some small trajectory departures as the perturbations become present or disappear, but some seconds after the system regain its close trajectory tracking; this can also be observed in Fig. 11, where also can be pointed out the effects

of takeoff and landing in the e_z plot. The Fig. 11 also shows cyclical bounded errors in e_x and e_y corresponding to the cyclical accelerations in the desired trajectory.

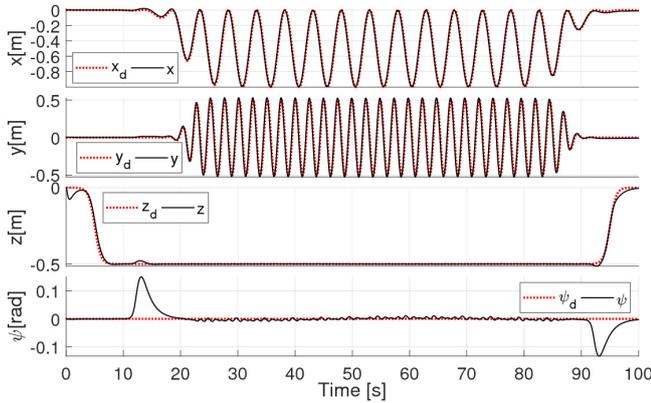


Fig. 10. Simulated and desired values of x , y , z , and ψ .

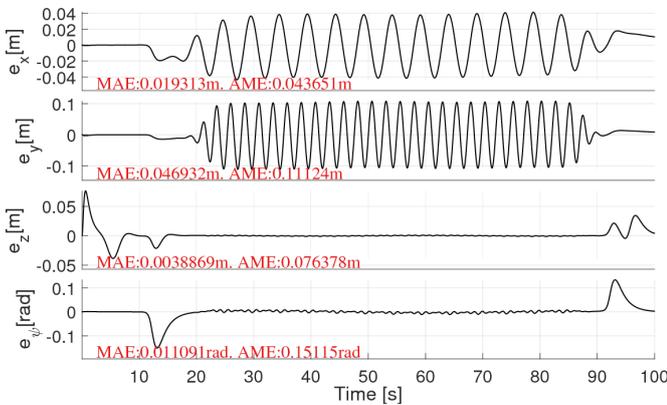


Fig. 11. Simulation tracking errors of x , y , z , and ψ .

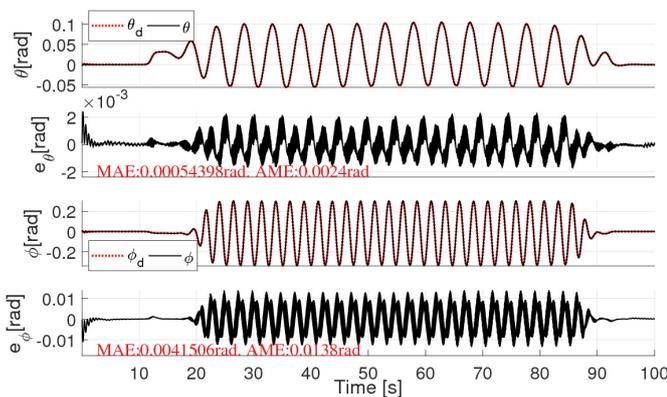


Fig. 12. Simulated and desired values, and tracking errors, of θ and ϕ .

The angular errors e_ϕ and e_θ are maintained close to zero, as can be observed in Fig. 12, even with ϕ close to 0.25rad, where the approximated model (13), used to calculate the control law, loses 3.1% of cosine accuracy and 24.74% of sine accuracy. Also, the applied PWM signals were kept under acceptable conditions for the physical system, staying slightly below to 50% of PWM duty for two rotors and near to 65% for the other two, and never going above 67.5% or below 42%, as shown in Fig. 13.

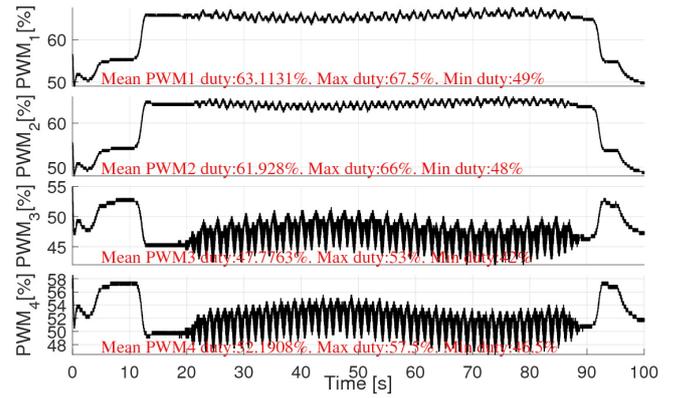


Fig. 13. Applied PWM duties.

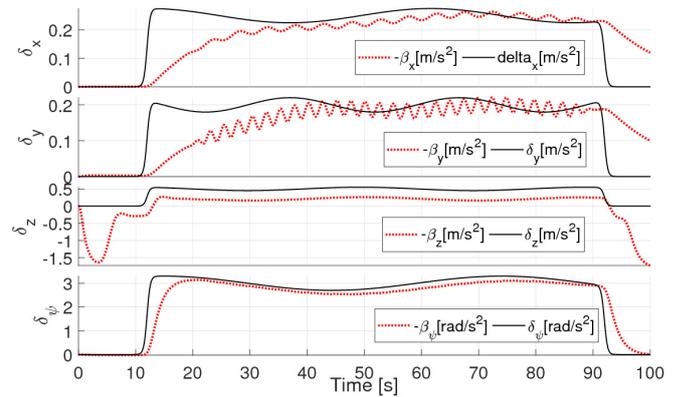


Fig. 14. Non-vanishing perturbations and extended states.

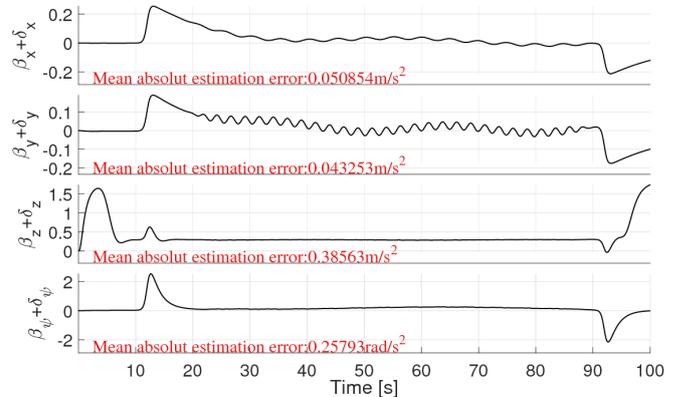


Fig. 15. Errors of non-vanishing perturbation estimation.

As can be observed in Fig. 14, the values of the β states approximate the values of the perturbations (17), presenting higher frequency variations due to the desired trajectory and errors introduced by model approximations, then Fig. 15 shows how the errors between extended states and perturbations grow in the takeoff and landing routines, and at the begin and end of perturbations, otherwise these errors appear due high acceleration movement.

5. REAL-TIME EXPERIMENTATION

The RT experimentation was carried out to evaluate the performance of the proposed control law in a physical system. As in Section 4, the control law (9) and the feedback lineari-

zation (14) were used. The system parameters and gain matrices in Section 4 are also used for the RT experimentation.

The experimental platform has three elements, an absolute localization system to measure the position ξ and the orientation ψ in the *NED* inertial frame, a quadrotor capable of performing RT evaluation of the proposed control law, and a computer/ground-station to read and filter the absolute localization system measurements, partially evaluate the control law and transmit control signals to the quadrotor.

The used absolute localization system is an *OptiTrack*[®] from *Natural Point*[®], with four *Flex-3*[®] cameras, an *OptiHub-2*[®], and the computer software *Motive*[®]. The measurements, from *Motive*, are read using C++ programming language under the Integrated Development Environment (IDE) *Microsoft Visual Studio Community 2017*[®]. Then, the computer/ground-station filters and partially evaluates the control law of the quadrotor, to transmit some control signals to the quadrotor using a *Xbee*[®] system. The used quadrotor is an *Ascending Technologies*[®] *Hummingbird*[®] shown in Fig. 16, this UAV receives some control signals, and reads Inertial Measurement Unit (IMU) measurements, then evaluates the remaining part of the control law and the linearization, and finally obtains the applied PWM signals.

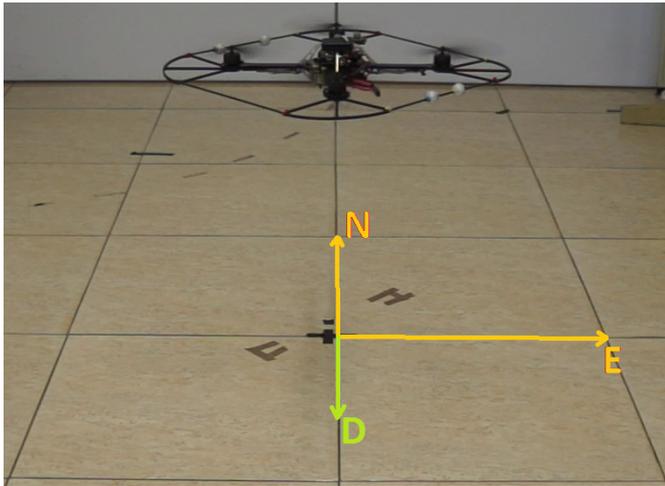


Fig. 16. Hummingbird quadrotor performing hover.

The developed software, for the RT experimentation, has two parts. The first one, at the ground-station, reads and filters the *OptiTrack* measurements, then evaluates the control law to obtain $\theta_d, \phi_d, \dot{\theta}_d, \dot{\phi}_d, \tilde{u}$, and $\tilde{\tau}_\psi$, and finally transmits those signals to the UAV. Then, the other part, at the quadrotor embedded computer, receives the control signals, reads $\theta, \phi, \dot{\theta}, \dot{\phi}$ from the embedded IMU, then evaluates the control law to obtain $\tilde{\tau}, \tau$, and u , and finally transforms the control signals into rotor thrusts and these into PWM signals. Fig. 17 presents a scheme that shows the RT program flow.

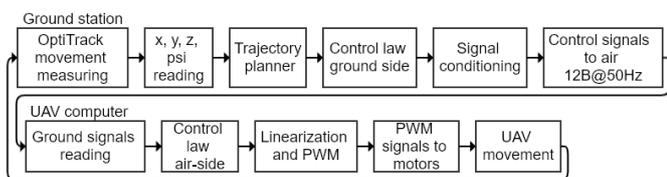


Fig. 17. Real-time software scheme.

5.1 Real-time experimentation for a quadrotor tracking a high acceleration trajectory

Real-time experimentation results are presented for the described experimental platform without intentionally added disturbance, using the high-acceleration desired trajectory (15) defined in Subsection 4.3.

5.1.1 Real-time experimentation results

The Figures 18 to 22 were obtained using the proposed control law (9) in RT experimentation, considering the desired trajectory (15). The Fig. 18 shows the desired and measured trajectories of the UAV in the *NED* inertial frame, and Fig. 19 shows the measured states of x, y, z , and ψ , compared to its desired counterparts, then Fig. 20 shows the tracking errors for those states.

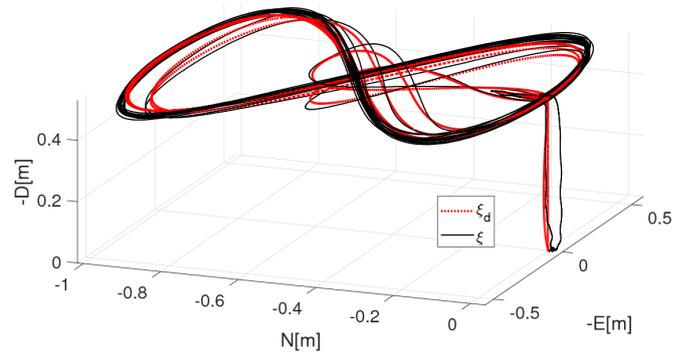


Fig. 18. Desired and measured trajectories in *NED* frame.

Subsequently, in Fig. 21 are shown the real-time-calculated desired values of θ and ϕ , their measured values, and the errors between them. Later, Fig. 22 shows the PWM duty cycles applied to the rotors.

5.1.2 Real-time experimentation discussion

The Figures 18 and 19 show the measured trajectory closely tracking the desired one, and as can be seen in Fig. 20 the tracking errors e_x and e_y are bounded under cyclical accelerations over 5.6m/s^2 . The height error e_z is kept bounded while tracking the trajectory, compensating properly the takeoff and landing ground-effect perturbation, and the orientation error e_ψ is also kept bounded. Having a scalar MAE for ξ 's three axes of 0.064m through all trajectory.

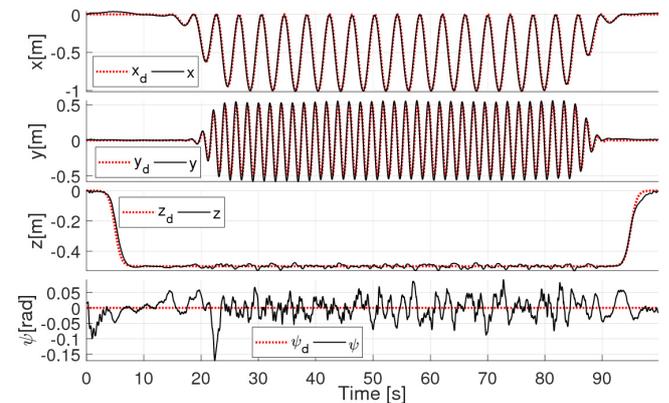


Fig. 19. Measured and desired values of x, y, z , and ψ .

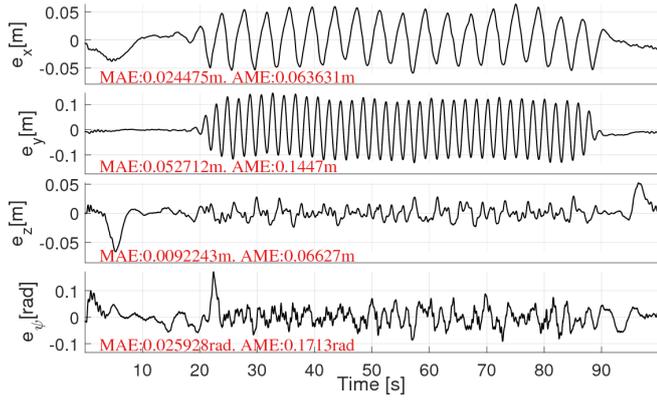


Fig. 20. Experimental tracking errors of x , y , z , and ψ .

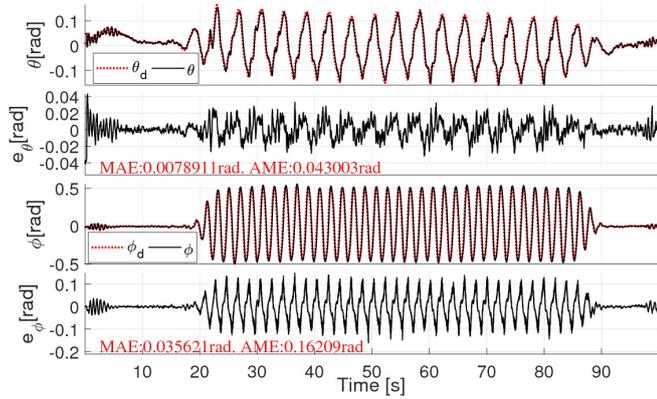


Fig. 21. Measured and desired values, and tracking errors, of θ and ϕ .

The Fig. 21 presents the measured angles ϕ and θ tracking their real-time-calculated desired counterparts, and the angular errors between them e_ϕ and e_θ , these errors are maintained close to zero even under angles over $\pi/6$ rad, where the model (13) and its feedback linearization (14) lose accuracy significantly. Also, the actuator-applied PWM signals are shown bounded in Fig. 22, but these signals show differences between actuators performances, particularly the PWM_4 signal shows higher control efforts up to 85.5% of PWM duty, this can be due to physical or wear down differences compensated by the control law.

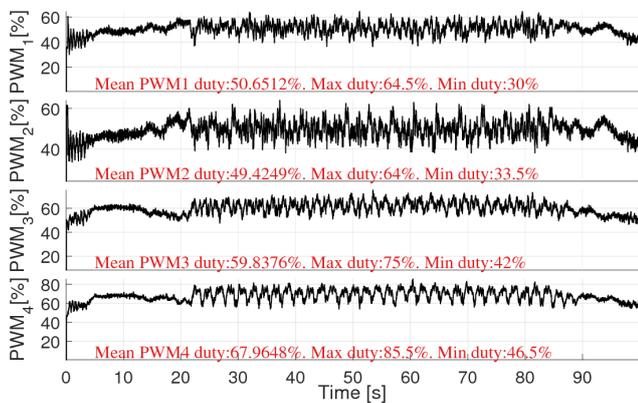


Fig. 22. Applied PWM duty cycles.

5.2 Real-time experimentation for a perturbed quadrotor tracking a trajectory

Real-time experimentation results are presented for the described experimental platform with intentionally added disturbance. The disturbance was provided by an electric fan in maximum operation, hand-carried to stay close to the UAV as shown in Fig. 23. The desired trajectory was elected to be the same as the used for the perturbed simulation (17) defined in Subsection 4.4.



Fig. 23. Hummingbird quadrotor under perturbations.

5.2.1 Real-time experimentation results

The Figures 24 to 29 were obtained using the proposed control law (9) in RT experimentation, using the setup described above. The Fig. 24 shows the desired and measured trajectories in the NED inertial frame, and Fig. 25 shows the measured and desired trajectories for x , y , z , and ψ , then Fig. 26 shows the tracking errors e_x , e_y , e_z , and e_ψ . The Fig. 27 shows the measured and desired values of θ and ϕ , and their tracking errors e_θ and e_ϕ . The Fig. 28 shows the behavior of the states β_i , $i = 1,2,3$, and finally Fig. 29 shows the applied PWM duties. All error plots show the MAE and AME.

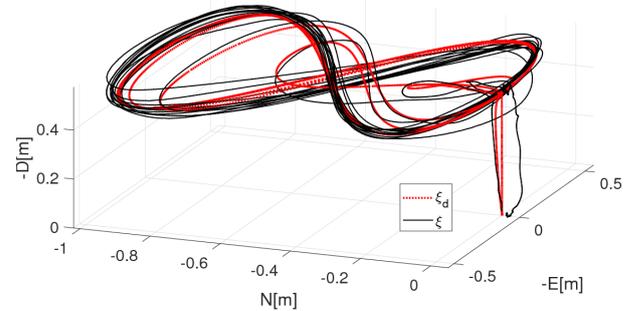


Fig. 24. Desired and measured trajectories in NED frame.

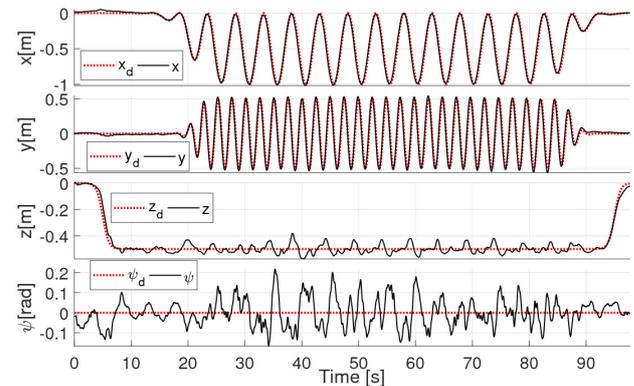


Fig. 25. Measured and desired values of x , y , z , and ψ .

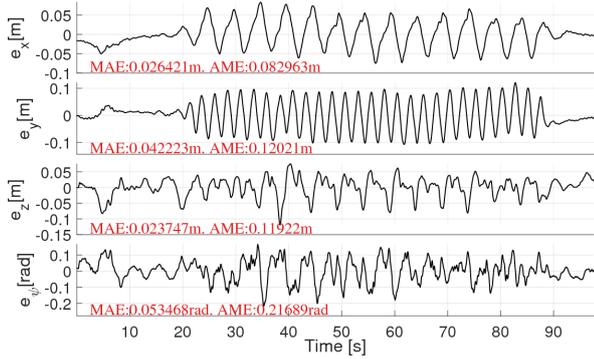


Fig. 26. Experimental tracking errors of x , y , z , and ψ .

5.2.2 Real-time experimentation discussion

The Fig. 24 shows the measured trajectory tracking the desired one, it can be observed some small displacements that do not affect the overall trajectory tracking, this also can be observed in Fig. 25, where the most notorious trajectory departures are observed in z and ψ plots. The Fig. 26 shows the tracking errors, where appear cyclical errors corresponding to the desired trajectory, all of these errors are bounded, and have a scalar MAE for ξ 's three axes of 0.061789m and a AME of 0.14049m, so it is considered that the desired trajectory is tracked appropriately under perturbations.

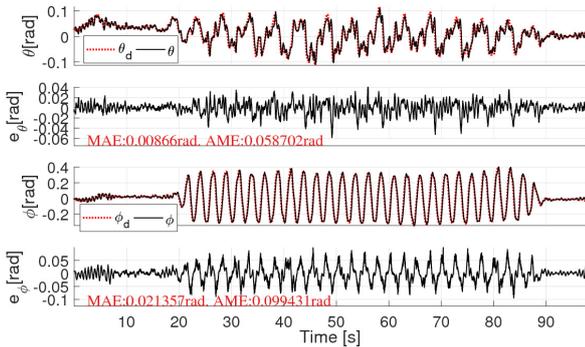


Fig. 27. Measured and desired values, and tracking errors, of θ and ϕ .

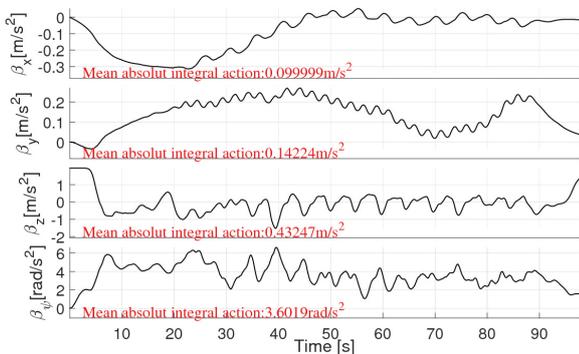


Fig. 28. Applied integral control actions.

The Fig. 27 shows an appropriate tracking of the desired angles θ_d and ϕ_d , where the tracking errors are kept close to zero even under perturbations and desired angles up to 0.4015rad. In Fig. 28 are observed the states β , that are meant to estimate and compensate the perturbations, where

can be seen that the non-vanishing perturbation was mostly applied parallel to the E axis, also the ground effect can be observed in the behavior of the β_z plot, in the β_ψ plot can be observed a strong compensation with a mean absolute value of 3.6019rad²/s that can be due to differences between rotor performances and the non-vanishing perturbation, in the β_x plot is observed that the integral compensation disappears after some time leaving only behaviors attributable to the desired trajectory and model approximations, and finally the β_y shows compensation for perturbations applied in $-E$ axis direction what corresponds appropriately to the disturbance applied.

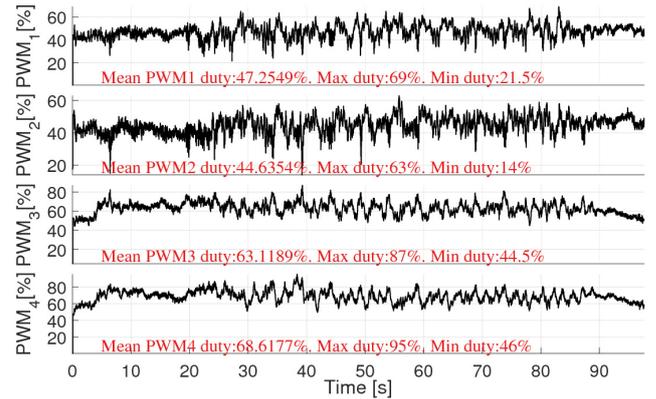


Fig. 29. Applied PWM duty cycles.

The Fig. 29 shows aggressive behaviors in the PWM duty cycles, applying up to 95% of the duty and some fast changing values, but these values do not go to saturation zone (0% or 100%) so the control signals were not trimmed.

5.3 General results discussion

Four result sets were presented, two for numeric simulations and two for RT experimentation, for both groups one simulation is for a high-acceleration trajectory and one for trajectory tracking while a non-vanishing perturbation is applied. The simulations results are considered positive as the errors are kept bounded and the control actions are maintained inside the physical capabilities of the UAV, this validate the viability of the proposed control law, and motivates the RT experimentation.

The RT results show that the proposed control law is applicable for physical systems, as the errors are maintained bounded, showing appropriate trajectory tracking for a high-acceleration desired trajectory or presence of non-vanishing perturbations. So, the proposed control law is considered as an applicable solution to high-acceleration desired trajectories or wind perturbed situations.

6. CONCLUSIONS

This paper presents a control law strategy design for a quadrotor, capable to maintain stability while tracking highly demanding trajectories or under perturbed scenarios, surpassing bounded vanishing and non-vanishing perturbations; perturbations that can be due to model approximations, parametric uncertainty, high acceleration movement, unmodelled dynamics, or wind conditions. The proposed controller is designed for a quadrotor dynamic model, using an integral-

backstepping-like technique that passes from position measurements to PWM signals, passing through desired angles. The error-correction actions in the proposed control law were bounded using the hyperbolic tangent function. The proposed control law applies integral control actions only to the perturbed equations of the system model, attending external forces or torques. Simulation and RT experimentation results for highly demanding trajectories and wind-perturbed scenarios are presented to show the advantages of the proposed control law, presenting well performance and maintained stability.

ACKNOWLEDGMENTS

This work was partially supported by the National Council for Science and Technology, CONACyT - Mexico, under the grant 254329.

REFERENCES

- Bouabdallah, S., and Siegwart, R. (2005). Backstepping and Sliding-mode Techniques Applied to an Indoor Micro Quadrotor. *Robotics and Automation (ICRA), 2005 IEEE International Conference on*, 2247-2252. IEEE. DOI:10.1109/ROBOT.2005.1570447.
- Bouabdallah, S., and Siegwart, R. (2007). Full control of a quadrotor. *2007 IEEE/RSJ International Conference on Intelligent Robots and Systems*, 153-158. IEEE. DOI:10.1109/IROS.2007.4399042.
- Chen, Y., and Pérez-Arancibia, N.O. (2016). Generation and real-time implementation of high-speed controlled maneuvers using an autonomous 19-gram quadrotor. *2016 IEEE International Conference on Robotics and Automation (ICRA)*, 3204-3211. IEEE. DOI:10.1109/ICRA.2016.7487489.
- Cheng, Z., Ma, Z., Sun, G., and Dong, H. (2017). Fractional order sliding mode control for attitude and altitude stabilization of a quadrotor UAV. *2017 Chinese Automation Congress (CAC)*, 2651-2656. IEEE. DOI:10.1109/CAC.2017.8243224.
- Cisneros, P.G., Hoffmann, C., Bartels, M., and Werner, H. (2016). Linear Parameter-Varying controller design for a nonlinear quad-rotor helicopter model for high speed trajectory tracking. *American Control Conference (ACC), 2016*, 486-491. IEEE. DOI:10.1109/ACC.2016.7524961.
- Dong, W., Gu, G.-Y., Zhu, X., and Ding, H. (2016). A high-performance flight control approach for quadrotors using a modified active disturbance rejection technique. *Robotics and Autonomous Systems*, 83, 177-187. DOI:10.1016/j.robot.2016.05.005.
- Garcia, A., Mattison, E., and Ghose, K. (2015). High-speed vision-based autonomous indoor navigation of a quadcopter. *Unmanned Aircraft Systems (ICUAS), 2015 International Conference on*, 338-347. IEEE. DOI:10.1109/ICUAS.2015.7152308.
- García-Carrillo, L.R., Dzul-López, A.E., Lozano, R., and Pégard, C. (2013). Modeling the Quad-Rotor Mini-Rotorcraft. *Quad Rotorcraft Control*, 23-34. Springer, London. DOI:10.1007/978-1-4471-4399-4_2.
- Guadarrama-Olvera, J.R., Rodríguez-Cortes, H., and Castro-Linares, R. (2014). Robust trajectory tracking control of a quadrotor helicopter. *2014 European Control Conference (ECC)*, 908-913. IEEE. DOI:10.1109/ECC.2014.6862570.
- Guerrero-Castellanos, J.F., Marchand, N., Hably, A., Lesecq, S., and Delamare, J. (2011). Bounded attitude control of rigid bodies: Real-time experimentation to a quadrotor mini-helicopter. *Control Engineering Practice*, 19(8), 790-797. DOI:10.1016/j.conengprac.2011.04.004.
- Huang, Y., Zheng, Z., Sun, L., and Zhu, B. (2017). Disturbance observer-based saturated control for a quadrotor landing on a vessel. *2017 13th IEEE International Conference on Control Automation (ICCA)*, 419-424. IEEE. DOI:10.1109/ICCA.2017.8003097
- Kanellakopoulos, I., and Krein, P.T. (1993). Integral-action nonlinear control of induction motors. *IFAC Proceedings Volumes*, 26(2), 117-120. DOI:10.1016/S1474-6670(17)49088-4
- Liu, S., Watterson, M., Tang, S., and Kumar, V. (2016). High speed navigation for quadrotors with limited onboard sensing. *Robotics and Automation (ICRA), 2016 IEEE International Conference on*, 1484-1491. IEEE. DOI:10.1109/ICRA.2016.7487284.
- Lopez, B.T., and How, J.P. (2017). Aggressive 3-D collision avoidance for high-speed navigation. *Robotics and Automation (ICRA), 2017 IEEE International Conference on*, 5759-5765. IEEE. DOI:10.1109/ICRA.2017.7989677.
- Loria, A., and Nijmeijer, H. (1998). Bounded output feedback tracking control of fully actuated Euler-Lagrange systems. *Systems & Control Letters*, 33(3), 151-161. DOI:10.1016/S0167-6911(97)80170-3.
- Madani, T., and Benallegue, A. (2006). Backstepping Control for a Quadrotor Helicopter. *International Conference on Intelligent Robots and Systems, Proceedings of the 2006 IEEE*, 3255-3260. IEEE. DOI:10.1109/IROS.2006.282433.
- Mellinger, D., Michael, N., and Kumar, V. (2012). Trajectory generation and control for precise aggressive maneuvers with quadrotors. *The International Journal of Robotics Research*, 31(5), 664-674. DOI:10.1177/0278364911434236
- Pretorius, A., and Boje, E. (2014). Design and modelling of a quadrotor helicopter with variable pitch rotors for aggressive manoeuvres. *IFAC Proceedings Volumes*, 47(3), 12208-12213. DOI:10.3182/20140824-6-ZA-1003.01586.
- Sepulchre, R., Jankovic, M., and Kokotovic, P.V. (1997). *Constructive Nonlinear Control* (1 ed.). Springer-Verlag, London. DOI: 10.1007/978-1-4471-0967-9.
- Skjetne, R., and Fossen, T.I. (2004). On integral control in backstepping: Analysis of different techniques. *American Control Conference, 2004. Proceedings of the 2004*, 1899-1904. IEEE. DOI:10.23919/ACC.2004.1386858.
- Vallejo-Alarcón, M.A., Velasco-Villa, M., and Castro-Linares, R. (2016). Quadcopter smooth-saturated robust backstepping control. *Mechatronics, Electronics and Automotive Engineering (ICMEAE), 2016 International Conference on*, 20-25. IEEE. DOI:10.1109/ICMEAE.2016.013.